Listing of Claims:

Claim 1 (currently amended): A memory arbiter, comprising:

a memory area; and

a memory controller coupled to the memory area, wherein the memory controller is configured to receive memory requests and corresponding priorities from a microprocessor, and wherein the memory controller is configured to continue to service current lower priority requests for a predefined first period if an incoming higher priority request is directed to a same page of memory as the current lower priority requests and to subsequently service the higher priority request for a predefined second period, and wherein the memory controller includes a first counter to monitor the predefined first period and a second counter to monitor the predefined second period.

Claim 2 (previously presented): The memory arbiter of claim 1, wherein the memory controller is configured to continue servicing any requests from a same agent as the lower priority request during the predefined first period.

Claim 3 (cancelled)

Claim 4 (previously presented): The memory arbiter of claim 2, wherein the memory controller is configured to eventually resume servicing any lower priority requests after the high priority request is processed.

Claim 5 (cancelled)

Claim 6 (currently amended): A method for servicing data in a computer system, comprising:

receiving a first memory request;

servicing the first memory request;

receiving a second memory request having a priority greater than the first memory request;

continuing to process the first memory request, and any other requests from a same agent as the first memory request, for a predefined first period if the second memory request is directed to a same page of memory as the first memory request, the predefined first period monitored by a first counter; and

servicing the second memory request, and any other requests having a priority greater than the first memory request, for a second predefined period after the first predefined period expires, the predefined second period monitored by a second counter.

Claim 7 (cancelled)

Claim 8 (cancelled)

Claim 9 (previously presented): The method of claim 6, further comprising:

resume eventually servicing any remaining lower priority requests after the higher priority request is processed.

Claim 10 (currently amended): A computer system, comprising:

a processor to initiate higher and lower priority memory requests;

a memory area; and

a memory controller coupled to the memory area, wherein the memory controller is configured to receive memory requests and corresponding priorities from the processor, and wherein the memory controller is configured to continue to service current lower priority requests for a predefined first period if an incoming higher priority request is directed to a same page of memory as the current lower priority requests and to subsequently service the higher priority request for a predefined second period, and wherein the memory controller includes a first counter to monitor the predefined first period and a second counter to monitor the predefined second period.

Claim 11 (previously presented): The computer system of claim 10, wherein the memory controller is configured to continue servicing any requests from a same agent as the lower priority request during the predefined first period.

Claim 12 (cancelled)

Claim 13 (previously presented): The computer system of claim 11, wherein the memory controller is configured to eventually resume servicing any lower priority requests after the high priority request is processed.

Claim 14 (cancelled)

Claim 15 (currently amended): An article comprising a machine-accessible medium having stored thereon instructions that, when executed by a machine, cause the machine to:

receive a first memory request;

service the first memory request;

receive a second memory request having a priority greater than the first memory request;

continue to process the first memory request, and any other requests from a same agent as the first memory request, for a predefined first period if the second memory request is directed to a same page of memory as the first memory request, the predefined first period monitored by a first counter; and

service the second memory request, and any other requests having a priority greater than the first memory request, for a second predefined period after the first predefined period expires, the predefined second period monitored by a second counter.

Claim 16 (cancelled)

Claim 17 (cancelled)

Claim 18 (previously presented): The article of claim 15, wherein the instructions, when executed by a machine, further cause the machine to:

resume eventually servicing any remaining lower priority requests after the higher priority request is processed.

Claim 19 (original): A memory arbiter, comprising:

a memory area; and

a memory controller coupled to the memory area, wherein the memory controller is configured to receive memory requests and corresponding priorities from a microprocessor, and wherein the memory controller is configured to interrupt servicing of higher priority requests after a predefined number are processed to process lower priority requests for a predefined period of time.

Claim 20 (original): The memory arbiter of claim 19, wherein the memory controller is configured to redefine the status of the higher priority requests to a lower priority status after a predefined number are processed.

Claim 21 (original): The memory arbiter of claim 19, wherein the memory controller is configured to reinstate the higher priority requests to it's initial priority status after the lower priority request is processed.

Claim 22 (original): The memory arbiter of claim 19, wherein the memory controller is configured to resume servicing higher priority requests after the predefined period expires.

Claim 23 (original): The memory arbiter of claim 19, further comprising:

a counter for monitoring the number of high priority requests.

Claim 24 (original): A method to service data in a computer system, comprising:

receiving and servicing a plurality of high priority memory requests;

receiving one or more memory requests having lower priority than the plurality of higher priority memory requests; and

interrupting servicing of higher priority requests after a predefined number is processed to process one or more lower priority requests for a predefined period of time.

Claim 25 (original): The method of claim 24, further comprising:

redefining the status of the higher priority requests to a lower priority status after a predefined number are processed.

Claim 26 (original): The method of claim 24, further comprising:

reinstating the higher priority requests to an initial priority status after the lower priority requests is processed.

Claim 27 (original): The method of claim 24, further comprising:

resuming servicing higher priority requests after the predefined period expires.

Claim 28 (original): A computer system, comprising:

a processor for initiating a higher and lower priority memory requests;

a memory area; and

a memory controller coupled to the memory area, wherein the memory controller is configured to receive memory requests and corresponding priorities from the processor, and wherein the memory controller is configured to interrupt servicing of higher priority requests after a predefined number is processed to process one or more lower priority requests for a predefined period of time.

Claim 29 (original): The computer system of claim 28, wherein the memory controller is configured to redefine the status of the higher priority requests to a lower priority status after a predefined number are processed.

Claim 30 (original): The computer system of claim 28, wherein the memory controller is configured to reinstate the higher priority requests to it's initial priority status after the lower priority request is processed.

Claim 31 (original): The computer system of claim 28, wherein the memory controller is configured to resume servicing higher priority requests after the predefined period expires.

Claim 32 (original): The computer system of claim 28, further comprising:

a counter to monitor the number of high priority requests.

Claim 33 (original): A machine readable medium having stored therein a plurality of machine readable instructions executable by a processor to service data, comprising:

instructions to receive and service a plurality of high priority memory requests;

instructions to receive one or more memory requests having lower priority than the plurality of higher priority memory requests; and

instructions to interrupt servicing of higher priority requests after a predefined number are processed to process one or more lower priority requests for a predefined period of time.

Claim 34 (original): The method of claim 33, further comprising:

instructions to redefine the status of the higher priority requests to a lower priority status after a predefined number are processed.

Claim 35 (original): The method of claim 33, further comprising:

instructions to reinstate the higher priority requests to an initial priority status after the lower priority requests are processed.

Claim 36 (original): The method of claim 33, further comprising:

instructions to resume servicing higher priority requests after the predefined period expires.